Evaluation of moments of quadratic forms and ratios of quadratic forms in normal variables

Magnus, J.R.; Pesaran, B.

Published in:
Computational Statistics

Publication date:
1993

Computational Statistics (1993) 8:21-37

Computational Statistics

Evaluation of Moments of Quadratic Forms and Ratios of Quadratic Forms in Normal Variables: Background, Motivation and Examples

J. R. Magnus

London School of Economics, Houghton Street, London WC2A 2AE, England and CentER for Economic Research, P.O. Box 90153, 5000 LE Tilburg, The Netherlands

B. Pesaran

Economics Division, Bank of England, Threadneedle Street, London EC2R 8AH, England

SUMMARY

Let x be normally distributed $N(\mu, \Omega)$, where Ω is positive definite and define the quadratic form $Q = x'Ax$ and ratio of quadratic forms $R = x'Ax/x'Bx$, where A is symmetric and B positive semi-definite. We present three routines: CUM calculates the first s cumulants and moments of Q; QRMOM calculates $E[R^s], E[R^s(a'x)]$ and $E[R^s(x'Cx)]$, where a is a vector and C a symmetric matrix, and also checks the existence of the expectations; PARINT is an auxiliary routine and works out all partitions of a given integer. This paper, the first in a series of four papers, gives some background and motivation and shows the reader how to use the routines. The other three papers are more technical and also present the full Fortran 77 code.

Keywords: Quadratic forms; Moments; Cumulants; Ratios of quadratic forms; Calculation of expectations; Tests for existence; Partitions of an integer.

1 Background and Motivation

Many tests of statistical hypotheses require a test-statistic which is a quadratic form in normal variables. In essence this is due to the fact that most estimators used in statistics and econometrics are (asymptotically) normally distributed, say $\hat{\theta} \sim N(\theta, V)$, so that the quadratic form $x'V^{-1}x$ with $x = \hat{\theta} - \theta$ follows a χ^2-distribution. The moments of a χ^2-distribution are known of course, but often these test-statistics take the form

$$Q = x'Ax$$

0943-4062/93/1/21-37 \$2.50

where $x \sim N(\mu, \Omega)$ and A is symmetric, so that Q does not follow a χ^2-distribution unless $AVA = A$.

The subroutine CUM calculates the cumulants and moments of Q. The routine is based on theory developed by Magnus (1978, 1979) and Don (1979) for the central case and Magnus (1986) for the general case. A different approach has recently been presented by Merckens and Wansbeek (1989) who use symbolic manipulation on the computer to form explicit expressions for such moments.

While the calculation of moments for quadratic forms is still relatively simple, the calculation of moments for ratios of quadratic forms presents difficulties of a different order. Such ratios arise in many applications. The simplest is perhaps the discrete first-order autoregression

$$y_t = \beta y_{t-1} + u_t \qquad (t = 1, ..., n),$$

where $y_0 = c$ (constant) and the $\{u_t\}$ are independent normal variates from $N(0,1)$. The least-squares estimator (also the maximum likelihood estimator) of β is

$$b = \frac{\sum_{t=2}^{n} y_t y_{t-1}}{\sum_{t=2}^{n} y_{t-1}^2}$$

which is a ratio of two quadratic forms, say $b = y'Ay/y'By$. There is a long history of interest in the moments of b. Expansions for its first two moments were first given by White (1961) and Shenton and Johnson (1965).

Hoque, Magnus and Pesaran (1988) considered the s-periods-ahead forecast, given by $\hat{y}_{n+s} = b^s y_n$ $(s = 1, 2, ...)$. The forecast bias is

$$E(\hat{y}_{n+s} - y_{n+s}) = E(b^s y_n)$$

and the mean-squared forecast error is

$$\begin{aligned} E(\hat{y}_{n+s} - y_{n+s})^2 &= \beta^{2s} \text{ var } (y_n) + E(b^{2s} y_n^2) \\ &-2\beta^s E(b^s y_n^2) + \sum_{j=0}^{s-1} \beta^{2j}, \end{aligned}$$

provided the expectations exist. Thus, while the moments of b are expectations of a ratio of two quadratic forms, the forecast bias and mean-squared forecast error are expectations of a ratio of quadratic forms times a linear form (y_n in the case of the forecast bias) or a quadratic form (y_n^2 in the case of the mean-squared forecast error). For more complicated autoregressive schemes we need more complicated expectations, but these are of the same type (see Magnus and Pesaran (1989, 1991)).

These obersvations have led us to a systematic study of statistics of the type

$$R = x'Ax/x'Bx$$

where $x \sim N_n(\mu, \Omega)$, A is symmetric and B positive semi-definite (possibly singular). The subroutine QRMOM can calculate

$$E[R^s] \quad , \quad E[R^s(a'x)] \quad , \quad E[R^s(x'Cx)]$$

where a is an $n \times 1$ vector and C a symmetric $n \times n$ matrix. The routine is based on theory developed by Magnus (1986, 1990), who provided exact expressions for the above expectations, which can be calculated numerically using one-dimensional integration, and also completely characterized the existence of those moments. Recently Smith (1988) followed a different approach and used zonal polynomials and invariant polynomials with multiple matrix arguments. See also Morin (1991) and Mathai (1991).

This paper is the first in a series of four papers (see also Magnus and Pesaran (1992a, b, c)) which present and discuss the subroutines CUM, QRMOM and an auxiliary routine PARINT, which works out all possible partitions of a given integer. All three routines will be available for use through the NAG library, probably Mark 16. The current paper gives some background and motivation, presents examples and shows the reader how to use the three routines without going into great technical deatil. The other three papers are more technical and also present the full Fortran 77 code.

2 The Subroutine PARINT

2.1 Purpose, Specification and Description

The subroutine PARINT works out all possible partitions of a given integer M. For example, the number 3 has three partitions, namely (1,1,1), (1,2) and (3). For arbitrary M, let n_i ($i = 1, ..., M$) denote the number of times that i appears in the partition. For $M = 3$ we have

$$(n_1, n_2, n_3) = (3, 0, 0) \ , \ (1, 1, 0) \ \text{ and } \ (0, 0, 1).$$

Thus, in order to find all partitions of M we need to obtain all possible vectors $(n_1, n_2, ..., n_M)$ such that the n_i are non-negative integers satisfying

$$n_1 + 2n_2 + ... + Mn_M = M.$$

Such partitions are used in a variety of situations. In particular they are used in calculating moments of products or ratios of quadratic forms in normal variables. The theory behind the code of PARINT was developed in Magnus (1978, 1979).

The specification (Fortran 77) is:

SUBROUTINE PARINT (M, MPARTS, MRDIM, MDIM, MR, IFAIL)

The integers M, MRDIM and MDIM are the inputs (unaltered on exit); the integer matrix MPARTS and the integers MR and IFAIL are the outputs. M is the integer for which the partitions are required. The output MR is the number of different partitions. The partitions are collected in a matrix MPARTS whose dimensions should be at least MR $\times$ M. The dimensions of MPARTS should be specified before entry. For example, the two lines

PARAMETER (MRDIM = 77, MDIM = 12)
DIMENSION MPARTS(MRDIM, MDIM)

allow calculation of all partitions of numbers up to 12. Thus we have MDIM $\geq$ M and MRDIM $\geq$ MR. The subroutine will check that these conditions are satisfied. The maximum value of M is 24 in which case we must specify MRDIM = 1575, MDIM = 24. To obtain partitions for M $>$ 24 the programme needs to be slightly altered, see Magnus and Pesaran (1992a) for details. IFAIL is an error indicator. If IFAIL $\neq$ 0 an error has occurred.

2.2 Time

CPU time needed for this subroutine is insignificant.

2.3 Accuracy

All calculations are carried out using integer arithmetic and are 100 % accurate.

2.4 Example

The programme below calculates all partitions of the number 7.

```
      PROGRAM PTEST
C
C     Calculates all partitions of 7
C     All partitions of numbers up to 12 are possible
C     For bigger numbers, the following statement should be changed
C
      PARAMETER(NPAR=77,NDIM=12)
      DIMENSION NPRTN(NPAR,NDIM)
      N=7
      CALL PARINT(N,NPRTN,NPAR,NDIM,NROW,IFAULT)
      WRITE(*,*) 'IFAULT=',IFAULT
      IF(IFAULT.NE.0) STOP
      WRITE(*,10) NROW,N
   10 FORMAT(/1X,I2,' partitions of ',I2,' found as follows:'/)
      DO 20 I=1,NROW
   20 WRITE(*,30)(NPRTN(I,J),J=1,N)
   30 FORMAT(1X,12I4)
      STOP
      END
```

The output will be "IFAULT = 0" followed by "15 partitions of 7 found as follows:" and the 15 × 7 matrix of Table 1.

Table 1. The partitions of the number 7.

7	0	0	0	0	0	0
5	1	0	0	0	0	0
4	0	1	0	0	0	0
3	2	0	0	0	0	0
3	0	0	1	0	0	0
2	1	1	0	0	0	0
2	0	0	0	1	0	0
1	3	0	0	0	0	0
1	1	0	1	0	0	0
1	0	2	0	0	0	0
1	0	0	0	0	1	0
0	2	1	0	0	0	0
0	1	0	0	1	0	0
0	0	1	1	0	0	0
0	0	0	0	0	0	1

3 The Subroutine CUM

3.1 Purpose, Specification and Description

The subroutine CUM (short for cumulants) calculates cumulants and moments of quadratic forms in normal variables. Let x follow an n-dimensional normal distribution with mean μ and variance-covariance matrix Ω (positive definite hence non-singular) and let A be an $n \times n$ symmetric matrix. Then define the quadratic form $Q = x'Ax$. The subroutine CUM obtains the first s cumulants and moments of Q. The routine is based on theory developed by Magnus (1978, 1979) for the central case and Magnus (1986) for the general case.

The specification (Fortran 77) is:

SUBROUTINE CUM (IMOM, N, LS, A, IEMU, EMU, VARLOW, RKUM, RMOM, WORK1, WORK2, WORK3, VEC, IFAULT).

Inputs are the integers IMOM, N, LS and IEMU, and the vectors EMU (of dimension at least N), A and VARLOW (both of dimension at least N(N+1)/2). All inputs, except possibly LS, are unaltered on exit. N is the dimension of x; there are no restrictions on N. The highest cumulant or moment required is denoted by LS. The programme will calculate all cumulants (and moments) from 1 to LS. Since CUM uses PARINT as an auxiliary routine, there are certain restrictions on LS. For LS $\leq$ 12 all is well. For LS $>$ 12 certain minor changes need to be made, see Magnus and Pesaran (1992a). IMOM and IEMU are indicators: If IMOM = 0 only the cumulants are calculated, if IMOM = 1 both cumulants and moments are calculated; if IEMU = 0 then $\mu = 0$, if IEMU $\neq$ 0 then $\mu \neq 0$. The vector EMU contains μ if $\mu \neq 0$. Storage should be allocated to EMU even when $\mu = 0$.

The covariance matrix Ω is entered through its Choleski decomposition. We write $\Omega = LL'$ where L is lower triangular with positive diagonal elements. (If Ω is not in this form the subroutine SEP described in Magnus and Pesaran (1992b) can be used to obtain it.) We enter L (VARLOW in the programme) as a vector $(l_{11}, l_{21}, l_{22}, l_{31}, l_{32}, l_{33}, \ldots)$. The symmetric matrix A is entered in the same way.

Outputs are the vector RKUM and RMOM (both of dimension at least LS) and the error indicator IFAULT. RKUM and RMOM contain the calculated cumulants and moments, respectively. If IFAULT $\neq 0$ an error has occurred.

The remaining vectors are work spaces.

3.2 Time

Using the VAX 6330 at the London School of Economics, typical CPU times for the double precision version are reported in Table 2 for different combinations of LS(4, 8 and 12) and N(10, 20, 30, 40 and 50). Other parameters were kept fixed at IMOM=1 and IEMU=1 as the variations in these did not affect the CPU time significantly. It should be pointed out that the effect of increase in LS is cumulative, i.e. when LS=4 all of the first 4 cumulants and moments are calculated while for LS=8 all first 8 cumulants and moments are evaluated by CUM.

Table 2. Typical CPU times for CUM

N	LS=4	LS=8	LS=12
10	0.30	0.32	0.47
20	1.07	1.47	1.98
30	3.74	5.17	6.39
40	10.43	13.24	16.35
50	23.81	29.52	35.26

3.3 Accuracy

In order to check the accuracy of calculations we used CUM to evaluate the first 4 cumulants and moments of the χ^2-distribution with 4 degrees of freedom. This was achieved by using an arbitrary 20×4 matrix R and by setting $A = R(R'R)^{-1}R'$ and $\Omega = I$. Setting IEMU=0 the first 4 cumulants and moments of $x'Ax$ were calculated.

The exact rth cumulant of a χ^2-distribution with ν degrees of freedom can be calculated using the formula

$$\kappa_r = \nu 2^{r-1}(r-1)! \tag{1}$$

while the moments of a χ^2-distribution with ν degrees of freedom can be worked out using its moment generating function which is

$$MGM(t) = (1-2t)^{-\nu/2}. \tag{2}$$

Table 3 allows the comparison between the exact results using (1) and (2) and the calculated values using CUM.

Table 3. Accuracy of cumulants and moments of a χ^2-distribution calculated by CUM

Cumulants			Moments	
r	Exact	Calculated	Exact	Calculated
1	4	4.00000000	4	4.0000000
2	8	8.00000000	24	24.00000000
3	32	32.00000000	192	192.00000000
4	192	192.00000000	1920	1920.00000000

Examining Table 3 shows that in this example an accuracy of at least eight decimal points is achieved.

3.4 Example

The test programme below considers the simple autoregression

$$y_t = \beta y_{t-1} + u_t \qquad (t = 1, ..., n),$$

where $\{u_t\}$ is a sequence of iid N(0,1) variables, and the start-up condition is

$$y_0 = CON.$$

We consider the expression

$$Q = \sum_{t=2}^{n} y_t y_{t-1}.$$

The vector $y = (y_1, y_2, ..., y_n)'$ follows an n-dimensional normal distribution with mean μ and variance matrix Ω which are easily calculated. Thus, Q can be written as a quadratic form, ie, $Q = y'Ay$. The programme CUMTEST calculates $E(Q^s)$.

We ran CUMTEST for fixed values of IMOM, n, LS and β, viz. IMOM=1, n=10, LS=12, β=0.8, and various values of IEMU and CON. The results are reported below.

```
PROGRAM CUMTEST
IMPLICIT REAL*8(A-H,O-Z)
PARAMETER(NDIM=50,ISDIM=12)
PARAMETER(NSYM=NDIM*(NDIM+1)/2)
DIMENSION A(NSYM),OMEGA(NSYM),EMU(NDIM)
DIMENSION WORK1(NSYM),WORK2(NSYM),WORK3(NSYM),VEC(NDIM)
DIMENSION RKUM(ISDIM),RMOM(ISDIM)
```

```
      DATA ZERO,HALF,ONE/0.0D0,0.5D0,1.0D0/
      WRITE(*,*)'TYPE IMOM IEMU BETA CON N LS'
      READ(*,*)IMOM,IEMU,BETA,CON,N,LS
      IFAIL=-1
      IF(N.GT.NDIM) GO TO 99
      N1=N-1
      NN=N*(N+1)/2
C
C     SET UP A, EMU, AND OMEGA
C
      DO 10 I=1,NN
10    A(I)=ZERO
      DO 20 I=1,N1
20    A(INX(I+1,I))=HALF
      EMU(1)=CON*BETA
      DO 30 I=1,N1
30    EMU(I+1)=BETA*EMU(I)
      DO 40 I=1,N
40    OMEGA(INX(I,I))=ONE
      BJ=ONE
      DO 50 J=1,N1
      BJ=BJ*BETA
      DO 50 I=1,N-J
      K=INX(I+J,I)
50    OMEGA(K)=BJ
C
C     END OF SETUP
C
      CALL CUM(IMOM,N,LS,A,IEMU,EMU,
     + OMEGA,RKUM,RMOM,WORK1,WORK2,WORK3,VEC,IFAIL)
      IF(IFAIL.NE.0) GO TO 99
      WRITE(*,*)'IMOM= ',IMOM
      WRITE(*,*)'IEMU= ',IEMU
      WRITE(*,*)'N= ',N
      WRITE(*,*)'BETA= ' ,BETA
      WRITE(*,*)'CON= ' ,CON
      DO 60 I=1,LS
60    WRITE(*,*)I,RKUM(I),RMOM(I)
99    IF(IFAIL.NE.0) WRITE(*,*)'IFAIL= ',IFAIL
      STOP
      END
```

Table 4. Results of CUMTEST.

IEMU	CON	s	RKUM	RMOM
0	0.0	1	16.12055071016092	16.12055071016092
0	0.0	2	306.3085759493393	566.1807311482089
0	0.0	3	13357.80135089808	32360.67240158577
0	0.0	4	922097.6202724892	2610052.863361274
0	0.0	5	86115423.36097491	272676783.0229518
0	0.0	6	10096621333.80519	34974282750.31927
0	0.0	7	1422398868643.269	5315037462786.988
0	0.0	8	233876735898685.5	933365546273876.6
0	0.0	9	4.3953925538393039E+16	1.8592781906007977E+17
0	0.0	10	9.2934525671189161E+18	4.1418349667651355E+19
0	0.0	11	2.1833279517629912E+21	1.0201821822256104E+22
0	0.0	12	5.6422857078669488E+23	2.7529152921246224E+24
1	0.0	1	16.12055071016092	16.12055071016092
1	0.0	2	306.3085759493393	566.1807311482089
1	0.0	3	13357.80135089808	32360.67240158577
1	0.0	4	922097.6202724892	2610052.863361274
1	0.0	5	86115423.36097491	272676783.0229518
1	0.0	6	10096621333.80519	34974282750.31927
1	0.0	7	1422398868643.269	5315037462786.988
1	0.0	8	233876735898685.5	933365546273876.6
1	0.0	9	4.3953925538393039E+16	1.8592781906007977E+17
1	0.0	10	9.2934525671189161E+18	4.1418349667651355E+19
1	0.0	11	2.1833279517629912E+21	1.0201821822256104E+22
1	0.0	12	5.6422857078669488E+23	2.7529152921246224E+24
1	1.0	1	17.51715245450299	17.51715245450299
1	1.0	2	350.0979349581493	656.9485650724493
1	1.0	3	16091.26081782378	39864.56678830199
1	1.0	4	1169960.889615146	3403882.798388269
1	1.0	5	114919700.0962483	375776361.3588519
1	1.0	6	14143612895.51915	50850688624.11257
1	1.0	7	2087274826806.565	8140756350807.859
1	1.0	8	358799600779190.8	1503795655510956.
1	1.0	9	7.0365109279532968E+16	3.1467114203194207E+17
1	1.0	10	1.5498134093139139E+19	7.3537587477915449E+19
1	1.0	11	3.7867669792761237E+21	1.8978387394739976E+22
1	1.0	12	1.0162683082285648E+24	5.3595923537454555E+24

4 The Subroutine QRMOM

4.1 Purpose, Specification and Description

The subroutine QRMOM (Quadratic Ratios of Moments) calculates moments of ratios of quadratic forms in normal variables. Again, x follows an n-dimensional normal distribution with mean μ and positive definite (hence non-singular) variance-covariance matrix Ω. Let A be symmetric and B symmetric positive semi-definite (hence possibly singular), both of order $n \times n$. We define the ratio $R = x'Ax/x'Bx$. The subroutine QRMOM can calculate for $s \geq 1$:

$$E[R^s] \tag{3}$$

$$E[R^s(a'x)] \tag{4}$$

and

$$E[R^s(x'Cx)] \tag{5}$$

where a is an $n \times 1$ vector and C a symmetric $n \times n$ matrix. The routine is based on theory developed by Magnus (1986, 1990). QRMOM also checks the existence of the moments using Theorems 1-3 of Magnus (1990).

QRMOM needs a routine which calculates eigenvalues and eigenvectors of a symmetric matrix and also a one-dimensional integration routine. Both routines are taken from the NAG library. Users who wish to substitute their own routines should consult Magnus and Pesaran (1992b) for details.

The specification (Fortran 77) is:

QRMOM(ICASE, NOBS, IS1, IS2, A, B, C, ELA, IEMU, EMU,
IOMEGA, OMEGA, ITEM, ISMAX, RESULT, ABSERR, IFAIL).

There are three indicator inputs: ICASE, IEMU and IOMEGA. ICASE = 1, 2, 3 determines which of the three cases (3), (4), (5) is calculated; IEMU = 0 if $\mu = 0$ and $\neq 0$ if $\mu \neq 0$; IOMEGA = -1, 1, 2 depending on how Ω (OMEGA) is supplied. If Ω itself is supplied then IOMEGA = 2, if L is supplied where $\Omega = LL'$ and L is lower triangular, then IOMEGA = 1; if L^{-1} is supplied, then IOMEGA = -1.

There are three other integer inputs: NOBS, IS1 and IS2. NOBS indicates the number of observations n. The programme requires that NOBS ≤ 50. This constraint can be lifted, see Magnus and Pesaran (1992b). IS1 and IS2 indicate the lowest and highest moment required in calculating (3), (4) or (5). QRMOM will calculate (3), (4) or (5) for all s satisfying IS1 $\leq s \leq$ IS2. We must have $1 \leq$ IS1 $\leq$ IS2 ≤ 12. The upper bound can also be lifted, see Magnus and Pesaran (1992b).

The input vectors A, B, C and OMEGA (corresponding to A, B, C and Ω) are of dimension at least NOBS *(NOBS+1)/2 and contain only the distinct elements of these symmetric matrices. For example, A is entered as $(a_{11}, a_{21}, a_{22}, a_{31}, a_{32}, a_{33}, ...)$. OMEGA may contain Ω, L or L^{-1}. In the latter two cases only the lower triangular elements are entered by row, as before.

The input vectors ELA and EMU (corresponding to a and μ) are of dimension at least NOBS.

There is no need to assign values to C and ELA when these are not used or to EMU when IEMU = 0, but storage should be allocated.

All inputs, except possibly IS1 or IS2, are unaltered on exit.

The output ISMAX reveals the maximum of s for which the required expectation exists (if the expectation exists for all s, we have set ISMAX = 100); the output ITEM tells us why, see Magnus and Pesaran (1992b) and Magnus (1990, Theorems 1-3). The output vectors RESULT and ABSERR contain the resulting expectations and the absolute errors made in the calculation. IFAIL is an error indicator.

4.2 Time

In order to work out the relationship between the CPU time and n and s we did some calculations on the VAX 6330 at the London School of Economics for different combinations of s (1 to 8) and n(10,20,30,40 & 50). In Table 5 the results of regressions of LOG(CPU) as a function of n and s for different values of ICASE and IEMU are reported. These regressions enable us to estimate the absolute CPU time (in seconds) for different combinations of n, s, ICASE and IEMU. Furthermore, if we differentiate the regression equations with respect to n or s we obtain the % increase in CPU time as a result of an increase in n or s. For example if we differentiate the regression equation for ICASE=1 and IEMU=0 with respect to s we have

$$\frac{\partial log(CPU)}{\partial s} = 2.172 + .003868 * n - 2 * .3701 * s - 2 * .0005736 * n * s + 3 * .0229 * s^2.$$

In particular, when n=40 and s=4, we obtain

$$\frac{\partial log(CPU)}{\partial s} = .2816.$$

This implies a growth in CPU time of about 28.2%. This should be compared with an increase in actual CPU time on the VAX from 380.2 seconds to 500.4 seconds which implies an increase of 31.6%. Hence using the relationships summarized in Table 5, once the timing of a particular combination of ICASE, IEMU, n and s is known, we can work out the marginal increase in CPU time. We estimate that the error in estimating the percentage change in CPU time is less than 15%.

Table 5. Results of least-squares regressions of LOG(CPU) as a function of n and s for different values of ICASE and IEMU

	ICASE=1 IEMU=0	ICASE=1 IEMU=1	ICASE=2 IEMU=1	ICASE=3 IEMU=0	ICASE=3 IEMU=1
Constant	-4.9046	-4.6367	-4.5340	-4.5108	-3.5408
n	0.2946	0.2967	0.3191	0.3182	0.2827
s	2.1720	2.0809	1.8911	1.8457	1.6686
n^2	-0.004397	-0.004623	-0.005783	-0.004956	-0.004182
ns	0.003868	0.004229	0.0070239	-	-
s^2	-0.3701	-0.3565	-0.3135	-0.2963	-0.2578
n^3	0.0000272	0.0000299	0.0000459	0.0000319	0.0000248
n^2s	-	-	-0.0000981	0.0000532	0.0000838
ns^2	-0.0005736	-0.0005799	-	-0.000583	-0.0008435
s^3	0.0229	0.0222	0.0176	0.0185	0.0167
$\bar{R}^2$	0.9951	0.9959	0.9957	0.9964	0.9960
SE of reg.	0.1393	0.1248	0.1302	0.1171	0.1169

4.3 Accuracy

The data statement in QRMOM sets ESPABS=1.0-5 (absolute error) and EPSREL=1.00-4 (relative error) in the calculation of the integrals. These can be changed in order to achieve different levels of accuracy.

In order to check the accuracy of calculations we used QRMOM to evaluate the first 4 moments of the F distribution with 4 and 16 degrees of freedom. This was achieved by using an arbitrary 20×4 matrix R and by setting $A = R(R'R)^{-1}R'$, $B = I - A$ and $\Omega = I$. Setting ICASE=1 and IEMU=0 the first 4 moments of $x'Ax/x'Bx$ were calculated. Since

$$F_{4,16} = 4(x'Ax/x'Bx),$$

we obtain

$$\mu_s' = E(F_{4,16})^s = 4^s E(x'Ax/x'Bx)^s.$$

The exact values for μ_s' can be calculated using the formula in Kendall and Stuart(1977, exercise 16.1 p. 423), which implies

$$E(x'Ax/x'Bx)^s = \frac{\Gamma(2+s)\Gamma(8-s)}{\Gamma(2)\Gamma(8)}. \tag{6}$$

Table 6 allows the comparison between the exact results using (6) and the calculated values using QRMOM for s=1 to 4.

Table 6. Accuracy of moments of an F distribution calculated by QRMOM

s	Exact	Calculated
1	2/7	0.28571429
2	1/7	0.14285714
3	4/35	0.11428571
4	1/7	0.14285714

Examining Table 6 shows that in this example an accuracy of at least eight decimal points is achieved.

4.4 Example

This test programme considers the simple autoregression

$$y_t = \beta y_{t-1} + u_t \qquad (t = 1, ..., n),$$

where $\{u_t\}$ is a sequence of iid N(0,1) variables, and the start-up condition is

$$y_0 = CON.$$

The least-squares estimator of β is given by

$$b = \frac{\sum_{t=2}^n y_t y_{t-1}}{\sum_{t=2}^n y_{t-1}^2}.$$

The vector $y = (y_1, y_2, ..., y_n)'$ follows an n-dimensional normal distribution with mean μ and variance matrix Ω which are easily calculated. Thus, b can be written as a ratio of two quadratic forms, ie, $b = y'Ay/y'By$.

The programme QRTEST will calculate

$$E(b^s),\ E(b^s y_n),\ E(b^s y_n^2)$$

depending on whether ICASE is set equal to 1, 2 or 3, respectively.

We ran QRTEST for fixed values of n and β, viz. n=10, β=0.8, and various values of ICASE, IEMU, CON, and s. The results are reported below.

```
      PROGRAM QRTEST
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(NDIM=50,ISDIM=12)
      PARAMETER(NSYM=NDIM*(NDIM+1)/2)
      DIMENSION A(NSYM),B(NSYM),C(NSYM),OMEGA(NSYM),
     + ELA(NDIM),EMU(NDIM),RESULT(ISDIM),ABSERR(ISDIM)
      DATA ZERO,HALF,ONE/0.0D0,0.5D0,1.0D0/
      WRITE(*,*)'TYPE ICASE IEMU N LS CON BETA'
      READ(*,*)ICASE,IEMU,N,LS,CON,BETA
      IOMEGA=1
      N1=N-1
      NN=N*(N+1)/2
      DO 10 I=1,NN
      A(I)=ZERO
      B(I)=ZERO
10    C(I)=ZERO
      C(NN)=ONE
      EMU(1)=CON*BETA
      DO 20 I=1,N1
      A(INX(I+1,I))=HALF
      B(INX(I,I))=ONE
      EMU(I+1)=BETA*EMU(I)
20    ELA(I)=ZERO
      ELA(N)=ONE
      DO 30 I=1,N
30    OMEGA(INX(I,I))=ONE
      BJ=ONE
      DO 40 J=1,N1
      BJ=BJ*BETA
      DO 40 I=1,N-J
      K=INX(I+J,I)
40    OMEGA(K)=BJ
      CALL QRMOM(ICASE,N,LS,LS,A,B,C,ELA,IEMU,EMU,
```

```
  +  IOMEGA,OMEGA,ITEM,ISMAX,RESULT,ABSERR,IFAIL)
     WRITE(*,*)'ITEM= ',ITEM
     WRITE(*,*)'ISMAX=',ISMAX
WRITE(*,*)'IFAIL=',IFAIL
IF(IFAIL.EQ.0) WRITE(*,*)LS,RESULT(1),ABSERR(1)
STOP
END
```

Table 7. Results of QRTEST.

ICASE	IEMU	CON	ITEM	LS	RESULT	ABSERR
1	0	0.0	2	1	0.6772271902076778	1.8204463788367862E-06
1	0	0.0	2	2	0.5343373869856963	2.0549211630979359E-08
1	0	0.0	2	3	0.4440335146183827	1.3728081080503790E-07
1	0	0.0	2	4	0.3885500362984481	4.7321832179260414E-07
1	1	1.0	2	1	0.6820430005371965	3.7507926660302086E-06
1	1	1.0	2	2	0.5356932893250592	4.7859018221943164E-08
1	1	1.0	2	3	0.4426936890953114	3.1135453514363818E-07
1	1	1.0	2	4	0.3840189937781694	9.5670752417186382E-07
2	1	1.0	3	1	0.1055212647023413	7.9533649246220010E-09
2	1	1.0	3	2	0.1040574973082275	3.1598280399186161E-07
2	1	1.0	3	3	0.1030010696327718	2.7465248903464417E-06
2	1	1.0	3	4	0.1023750660410675	6.5580100358892616E-06
3	0	0.0	4	1	2.378528327169972	5.5431501324961895E-05
3	0	0.0	4	2	2.223537861504993	7.3945403835994983E-07
3	0	0.0	4	3	2.156178253010546	4.9334392491888219E-06
3	0	0.0	4	4	2.162236773773190	1.3716232648755862E-05
3	1	1.0	4	1	2.376219657733249	1.0552110800027964E-04
3	1	1.0	4	2	2.200222882843882	1.5712239265410085E-06
3	1	1.0	4	3	2.111558376997302	1.0535305223719664E-05
3	1	1.0	4	4	2.092669295338627	2.6810456817462994E-05

References

Don, F.J.H. (1979). The expectation of products of quadratic forms in normal variables. Statistica Neerlandica, 33, 73-79.

Hoque, A., Magnus, J.R. and Pesaran, B. (1988). The exact multi-period mean-square forecast error for the first-order autoregressive model. Journal of Econometrics, 39, 327-346.

Magnus, J.R. (1978). The moments of products of quadratic forms in normal variables. Statistica Neerlandica, 32, 201-210.

Magnus, J.R. (1979). The expectation of products of quadratic forms in normal variables: the practice. *Statistica Neerlandica, 33, 131-136.*

Magnus, J.R. (1986), The exact moments of a ratio of quadratic forms in normal variables. *Annales d'Economie et de Statistique, 4, 95-109.*

Magnus, J.R. (1990). On certain moments relating to ratios of quadratic forms in normal variables: further results. *Sankhyā, Series B, 52, 1-13.*

Magnus, J.R. and Pesaran, B. (1989). The exact multi-period mean-square forecast error for the first-order autoregressive model with an intercept. *Journal of Econometrics, 42, 157-179.*

Magnus, J.R. and Pesaran, B. (1991). The bias of forecasts from a first-order autoregression. *Econometric Theory, 7, 222-235.*

Magnus, J.R. and Pesaran, B. (1992a). The evaluation of cumulants and moments of quadratic forms in normal variables (CUM): technical description. *Computational Statistics, this issue.*

Magnus, J.R. and Pesaran, B. (1992b). The evaluation of moments of ratios of quadratic forms in normal variables and related statistics (QRMOM): technical description. *Computational Statistics, this issue.*

Magnus, J.R. and Pesaran, B. (1992c). CUM, PARINT and QRMOM - Evaluation of moments of quadratic forms and ratios of quadratic forms in normal variables: Fortran 77 Code. CentER, Tilburg University.

Mathai, A.M. (1991). On moments of ratios of quadratic expressions. Discussion paper, Department of Statistics, McGill University, Montreal.

Merckens, A. and Wansbeek, T. (1989). Formula manipulation in statistics on the computer: Evaluating the expectation of higher-degree functions of normally distributed matrices. *Computational Statistics & Data Analysis, 8, 189-200.*

Morin, D. (1991). On ratios of quadratic forms in normal variables. Discussion Paper, Department of Decision Sciences and M.I.S., Concordia University, Montreal.

Numerical Algorithms Group (1984). *NAG Fortran Library Manual,* Mark 11, Naglib, London.

Shenton, L.R. and Johnson, W.L. (1965). Moments of a serial correlation coefficient. *The Journal of the Royal Statistical Society,* Series B, 27, 308-320.

Smith, M.D. (1988). On the expectation of a ratio of quadratic forms in normal variables. Discussion paper 88-02, Department of Econometrics, University of Sydney.

White, J.S. (1961). Asymptotic expansions for the mean and variance of the serial correlation coefficient. *Biometrica, 48, 85-94.*